Alizain Charania 09/20/2015

#beginning point: 100% correct 100% W/E Free Trials

Resource Metrics	Your Code's Starting Statisitcs
Static code size	39
Dynamic execution length	184
Storage (in registers, static memory and stack)	40

O1:
Change slt $15, $8, $9 to beq $8, $9, End
This will save 1 register ($15) and also reduce the dynamic execution length by 1, each loop.
O2:
Change slti $16, $14, 1 to bne $14, $0, End
This will save 1 register ($16) and also reduce the dynamic execution length by 1, each loop.
O3:
Remove and $17, $15, $16 and beq $17, $0, End
With the optimizations done in O1 and O2, I can remove these two lines. This will save 1 register ($17) and also reduce the dynamic execution length by 2, each loop. It reduced the static code size by 2.
O4:
Remove addi $5, %0, 4
Preventing multiple initialization of the counter, i. This will save 1 register ($5) and reduce the static code size by 1.
O5:
Change $7 to $8 in slt $7, $4, $6 and beq $7, $0, Else1
Re-using of the temporary registers. This will reduce 1 register ($7).
O6:
Add j End1 in the If3 block of code
This will act as a break point. Once I find the first year when Harry and Sally met, I can jump to the End and terminate the program. This will save many useless looping after finding the correct answer. This will increase the static code by 1 but will reduce the dynamic length greatly, on average.
O7:
Remove add $14, $0, $8
I can remove a static code line and not store the final year in a temporary register. This will save 1 register ($14) and also reduce couple static code size.
O8:
Change addi $14, $0, 0 to addi $2, $0, 0
Directly initializing the answer to 0 will save 1 register ($14).
O9:
Remove bne $14, $0, End
Re-optimize optimization O2 as the break point optimization in O6 will handle the check for the termination of the loop once the year is found. This will be very efficient as it will reduce 1 static code line but will reduce the re-execution of conditional line with only 1 execution of the break statement during dynamic execution.
BF1:
Add End1 block
This block will execute the line year = begin year if and only if the year is found or else the code will go to End and output the pre-initialized year = 0.

O10:
Change $19 to $10 in addi $19, $10, 4 and lw $12, Harry($19)
This will reduce 1 register ($19) and use $10 more efficiently instead.
BF2:
Change addi $10, $10, 8 to addi $10, $10, 4
This was needed to be done to fix the bug created in optimization O10.
O11:
Change $21 to $11 in addi $21, $11, 4 and lw $13, Sally($21)
This will reduce 1 register ($21) and use $11 more efficiently instead.
BF3:
Change addi $11, $11, 8 to addi $11, $11, 4
This was needed to be done to fix the bug created in optimization O11.
O12:
Change $18 to $4 in lw $18, Harry($10) and bne $8, $18, If2
Re-using the temporary registers. This will save 1 register ($18).
O13:
Change $20 to $6 I lw $20, Sally($11) and bne $8, $20, If2
Re-using the temporary registers. This will save 1 register ($18).

Resource Metrics	Your Code's Starting Statisitcs
Static code size	35
Dynamic execution length	124
Storage (in registers, static memory and stack)	32